Using OLSR Multipoint Relays (MPRs) to estimate node positions in a Wireless Mesh Network

Erwan Ermel, Paul Mühlethaler

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Using OLSR Multipoint Relays (MPRs) to estimate node positions in a Wireless Mesh Network

Erwan Ermel , Paul Muhlethaler

No 6072

Décembre 2006

THÈME 1

Rapport de recherche

ISSN 0249-6399

Using OLSR Multipoint Relays (MPRs) to estimate node positions in a Wireless Mesh Network

Erwan Ermel [*] , Paul Muhlethaler [†]

Thème 1 — Réseaux et systèmes
Projet HIPERCOM

Rapport de recherche n˚6072 — Décembre 2006 — 17 pages

Abstract: In this paper we address the problem of nodes localization in wireless heterogeneous networks, focussing particularly on **anchor** selection methods to estimate position. The **Optimized Link State Routing protocol (OLSR)** [1] uses special nodes called **Multipoint Relay (MPR)** nodes to broadcast control messages within the network. We propose a novel approach based on using these Multipoint Relay (MPR) nodes as anchor nodes to estimate nodes positions. We evaluate its performance by simulation and compare it to other selection techniques such as convex hull selection and greedy selection.

Key-words: Localization, Positioning, Wireless Mesh Network (WMN), Selection Methods, Optimized Link State Routing protocol (OLSR), Multipoint Relay (MPR), Convex Hull.

(Résumé : tsvp)

[*] Laboratoire d'Informatique de Paris 6, Université Pierre et Marie Curie
[†] HiPERCOM project, INRIA

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
Téléphone : 01 39 63 55 11 - International : +33 1 39 63 55 11
Télécopie : (33) 01 39 63 53 30 - International : +33 1 39 63 53 30

Utilisation des relais multipoint pour estimer la position des noeuds dans un réseau sans fil maillé

Résumé : Dans ce papier, nous traitons du problème de la localisation dans un réseaux sans fils hétérogène et plus particulièrement de la sélection de noeuds **ancre** à partir desquels on estime la position. Le protocole de routage **OLSR (Optimized Link State Routing protocol)** [1] utilise des noeuds spéciaux appelés relais multipoint **(MPR Multipoint Relay)** pour diffuser les messages de controle dans le réseau. Nous proposons une approche nouvelle basée sur ces relais multipoint comme noeuds ancre pour estimer la position des noeuds. Nous évaluons sa performance par simulation et comparons celle-ci à d'autres techniques de sélection telles que la sélection de l'enveloppe convexe ou encore l'approche gloutonne.

Mots-clé : Localisation, positionnement, réseau mesh, méthode de sélection, Optimized Link State Routing protocol (OLSR), relais multipoint, enveloppe convexe

1 Introduction

With the tremendous growth in the number of wireless devices over the last decade, many new fields of research have opened up. In such conditions and in such wireless networks, the position of network nodes now represents valuable information. At the same time, being able to maintain network connectivity using network nodes as relays can be very useful to develop new applications.

A node can estimate its position by using a GNSS (Global Navigation Satellite System) like GPS [2] or Galileo. It gives good position precision but GNSS chips are still costly for low cost wireless devices, and this position estimation method does not work in an indoor environment. A node with no self-location capabilities can estimate its position by several means. Neighbor nodes can be used as "anchors" to estimate its position. To estimate its position, a node can evaluate its distance to the anchors with several methods like the RSSI (Received Signal Strength Indicator) [3, 4], Time of Arrival (TOA) [5] or by a connectivity approach [6, 7]. But estimating a position in a greedy way (all anchor nodes are used in the position estimation method) is not a good solution. Ermel et al [8] show that with a simple convex hull selection among the anchors, the precision of the estimated position is enhanced by up to 20% for configurations with the same number of anchors.

In addition to node localization, a crucial task in wireless network is to maintain its connectivity. To do so the ad hoc network has to use a routing protocol. Many studies have been carried out on routing protocols for Mobile Ad Hoc Networks (MANETs) especially at the IETF in the working group MANET. Two kinds of routing protocols have been proposed: reactive protocols and proactive protocols. In reactive protocols, the routes are built only upon demand. The source node wishing to obtain a route to a destination floods the network with a request packet. When the destination node receives the request packet it responds to the source node and the path used by the request packet will be the route from the source to the destination node. AODV [9] and DSR [10] are examples of such protocols. On the other hand, proactive protocols maintain the knowledge of the network topology through the exchange of periodic packets. In such protocols the main issue is to reduce the control overhead. When a link state protocol is used, it is important to optimize the broadcast in the network. OLSR and TBRPF [1, 11] are examples of such protocols. In this article we intend to show that the genuine optimization of OLSR can be used to estimate a node's position. We present a novel anchor selection method to estimate a position based on OLSR Multipoint Relay (MPR) nodes. This method is based on node connectivity. Details on MPR nodes will be given in Section 3.

The paper is structured as follows. Section 2 presents the assumptions and definitions used in this paper. Section 3 details the MPR nodes selection method. Section 4 presents our simulation results. Section 5 concludes the paper.

2 Definitions and assumptions

In this paper, we will not use distance measurements as they can be derived, for instance, from signal strength measurements to compute a node's position. A node only exploits its neighboring nodes connectivity to estimate its position; additionally it can know the position of its neighbor nodes. A node will generally use special neighbor nodes called anchors to estimate its position. In this paper we limit our approach to selecting only 1-hop anchors i.e. anchors who are neighboring nodes.

Two types of nodes are considered in a heterogeneous wireless network: Self-Locating Nodes (SLN) which are nodes with self-locating capabilities e.g. nodes with a GPS, and Simple Nodes (SN) which have to estimate their position by other means. SN (simple nodes) estimate their position by a simple centroid formula (1), in which all the nodes are given the same weight.

$$(X_{est}, Y_{est}) = \left(\frac{\sum_{i=1}^{n} X_i}{n}, \frac{\sum_{i=1}^{n} Y_i}{n} \right). \tag{2.1}$$

When a node selects anchors to compute its position formula (1) computes the centroid only of these anchors.

Let S be a *simple* node. Let S_{est} be the estimated position of the node S, and S_{real} be the coordinates of its real location. Note that S_{real} information is only used by simulations to evaluate the precision of our algorithm. We can not possess such information in a real world as it represents what we are looking for. Let R_{max} be the maximum theoretical transmission range of S. We define the accuracy of the node position C_{acc}, as a function of the localization error represented by the distance between S_{real} and S_{est}:

$$C_{acc} = 1 - \frac{\|(S_{real}, S_{est})\|_2}{R_{max}} \tag{2.2}$$

By definition, $0 \leq C_{acc} \leq 1$. Self-locating nodes (SLN) with accurate coordinates have a position accuracy of 1. Conversely, simple nodes (SN), which have to estimate their own position, have a position accuracy of $C_{acc}/in[0, 1[$.

We also define the theoretical mean number of neighbor nodes N_{th} as:

$$\mathcal{N}_{th} = \frac{\mathcal{N}}{\mathcal{A}} \mathcal{A}_{coverage} \tag{2.3}$$

where $\mathcal{N}$ is the total number of nodes, $\mathcal{A}$ is the simulation area and $\mathcal{A}_{coverage}$ is the radio coverage area.

3 Anchor selection methods

The goal of this paper is to enhance the accuracy of an estimated position by selecting anchors that are likely to improve the position estimation process. In this section we detail our MPR node selection method and two other approaches: the greedy and the convex hull methods.

3.1 MPR Selection

In Wireless Mesh Networks (WMN), the medium is usually shared thus: when a packet is flooded, the same packet is sent many times to the same receiver. Not only is this a waste of bandwidth but also, since the load of broadcast packets is increased in the network, it may increase the collision rate and the actual packet delivery may then be decreased. The multipoint relay technique is used to reduce the overhead induced by transmitting of broadcast packets. The concept of multipoint relay was first introduced in [12] for HiPERLAN type 1 and the multipoint relay optimization is the core optimization of OLSR [13]. The main idea of the multipoint relay optimization is that only a subset of neighbors has to relay a flooded packet that has been flooded. It can be easily understood that if a conveniently chosen subset of one's neighbor nodes can relay a flooded packet to all one's 2-hop neighbors; then the relay of these nodes will be sufficient to ensure the proper delivery of the packet to the node m's 2-hop neighbors, see Figure 1. This subset of nodes is called the multipoint relay set of node m; of course the smaller the number of nodes in the multipoint relay set, the greater the optimization.

Multipoint relay optimization must be repeated recursively when the packet is flooded. At each hop a flooded packet is relayed by the next hop multipoint relay set. Of course an already transmitted packet is not retransmitted twice; this is controlled by a duplicate table.

The interesting point is that the notion of multipoint relay is deeply embedded in the OLSR protocol. To maintain the knowledge of the network topology OLSR uses two kinds of control. The first kind of packet called "hello" is used to build the neighborhood. The second kind of packet called "TC" is used by each node to disseminate the neighborhood within the network. The two main *OLSR* functionalities, Neighbor Discovery and Topology Dissemination, are now detailed.

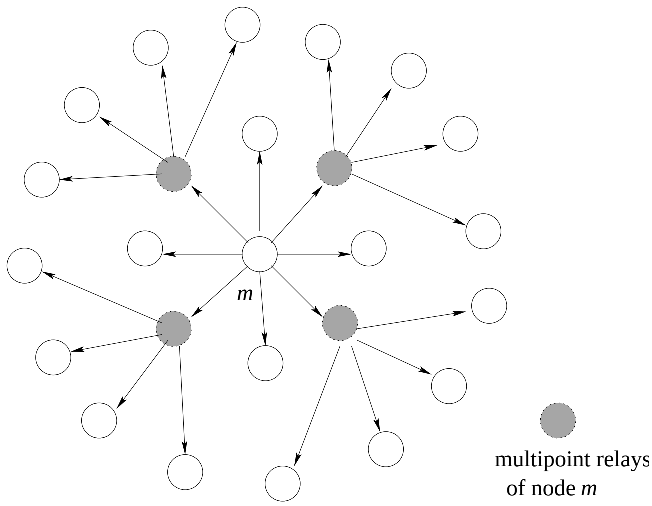

Figure 1: Multipoint relays of node m

3.1.1 Neighbor Discovery

Each node must detect the neighbor nodes with which it has a direct link. For this, each node periodically broadcasts *Hello* messages, containing the list of neighbors known to the node and their link status. The link status can be either *symmetric* (if communication is possible in both directions), *asymmetric* (if communication is only possible in one direction), *multipoint relay* (if the link is symmetric and the sender of the *Hello* message has selected this node as a *multipoint relay*), or *lost* (if the link has been lost). The *Hello* messages are received by all 1-hop neighbors, but are not forwarded. They are broadcasted once per refreshing period called the "*HELLO_INTERVAL*". Thus, *Hello* messages enable each node to discover its 1-hop neighbors, as well as its 2-hop neighbors. This neighborhood and 2-hop neighborhood information has an associated holding time, the "*NEIGHBOR_HOLD_TIME*", after which it is no longer valid.

On the basis of this information, each node independently selects its own set of *multipoint relays* among its 1-hop neighbors in such a way that all 2-hop neighbors of m have *symmetric* links with $MPR(m)$. This means that the *multipoint relays* cover (in terms of radio range) all 2-hop neighbors (Figure 1). The *multipoint relay* set is computed whenever a change in the 1-hop or 2-hop neighborhood is detected. In addition, each node m maintains its "*MPR selector set*". This set contains the nodes which have selected m as a *multipoint relay*. Node m only forwards broadcast messages received from one of its *MPR selectors*.

It can be noticed that “Hello” messages are ideal to carry a node’s position. This information can simply be added at the end of the messages.

3.1.2 Topology Dissemination

Each node of the network maintains topological information about the network obtained by means of *TC (Topology control)* messages. Each node m selected as a *multipoint relay*, broadcasts a *TC* message at least every “*TC_INTERVAL*”. The *TC* message originated from node m declares the *MPR selectors* of m. If a change occurs in the *MPR selector* set, the next *TC* can be sent earlier. The *TC* messages are flooded to all nodes in the network and take advantage of *MPRs* to reduce the number of retransmissions. Thus, a node is reachable either directly or via its *MPRs*. This topological information collected in each node has an associated holding time “*TOP_HOLD_TIME*”, after which it is no longer valid.

The neighbor information and the topology information are refreshed periodically, and they enable each node to compute the routes to all known destinations. These routes are computed with Dijkstra’s shortest path algorithm [14]. Hence, they are optimal as concerns the number of hops. The routing table is computed whenever there is a change in neighborhood or topology information.

OLSR’s routes which are optimal in terms of the number of hops are constructed with MPR nodes. Thus, we can intuitively understand that the MPR nodes of a node A will optimally “surround” A and that they are good candidates for being anchors to estimate a position.

3.1.3 The proposed heuristic for MPR selection

Finding the smallest (in term of number of nodes) multipoint relay set has been shown to be NP hard [15] however efficient heuristics exist. The first proposed heuristic is described in [16] and uses a greedy approach. This algorithm can be simply described as follows. Let us consider a given node A and its 2-hop neighbors N_{2h}; the 2-hop neighbors of A are the neighbors of $A's$ neighbors who are not already $A's$ neighbors. The algorithm first selects as multipoint relays the neighbors who are the only path to a given 2-hop neighbor. When these nodes are selected the 2-hop neighbors that are reachable through these nodes are removed from the 2-hop neighbors set N_{2h}. The algorithm then select the neighbor which covers the highest number of 2-hop neighbors in N_{2h}. In the case of equality between two nodes a simple tie is used. When a node is selected, the 2-hop neighbors of N_{2h} that are reachable through these nodes are removed from N_{2h}. The process is carried on until there is no node left within N_{2h}.

An example of such a selection is presented in Figure 2. Node B is first selected because it is the only path to a 2-hop neighbor. Then nodes D and C which cover four nodes in the two neighborhood are selected. Then node E which covers three nodes in the 2-hop neighborhood of A is selected. Node F allows the two remaining nodes in the hops neighborhood of A to be covered.

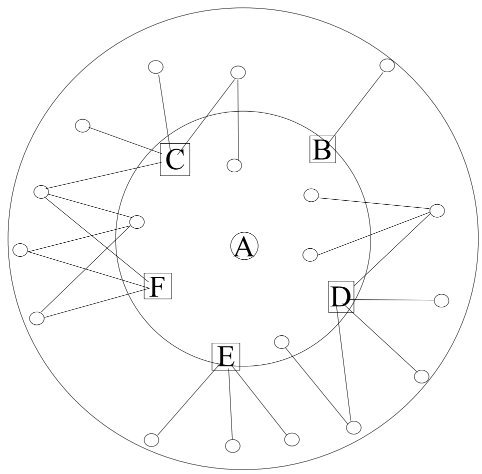

Figure 2: Greedy algorithm for the MPR selection.

3.2 Greedy and Convex Hull anchor selection methods

To evaluate the performance of our MPR selection method, we will compare its performances to two other approaches: the greedy and the convex hull selection methods.

3.2.1 Greedy approach

No selection is made among the nodes, all neighbor nodes are used in the position estimation process. This approach is called the greedy approach.

3.2.2 Convex Hull

The main idea of using a convex hull as a selection method among nodes is to choose only nodes which are at the greatest distance from anchors. In the plane, the convex hull can be visualized as the shape assumed by a rubber band that has been stretched around a set of points and released to conform as closely as possible all neighbor nodes. As the position estimation process is based on trilateration, the further apart the anchors are, the better the accuracy of the estimated position will be, see Figure 3).

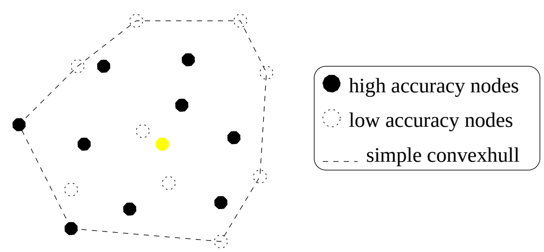

Figure 3: Convex hull: simple convex hull considers the distance metric to elect hull nodes.

To solve this geometrical problem, computational geometry algorithms have to be used e.g. Gift Wrapping (Jarvis's March) [17], [18, 19, 20, 21, 22, 23]. All these approaches are well detailed in [24, 25, 26]. We choose the Quick Hull algorithm [25] as our convex hull selection method. In our convex hull selection we de not take into account the position accuracy of the neighbor nodes.

In previous work [8], we introduced the convex hull as a selection method in wireless networks. The resulting position accuracy using a convex hull node selection method is greater than a greedy scheme by up to 20%.

3.3 Complexity of the selection methods

Let us denote by n the number of a node's neighbors. To select MPR nodes the main cost consists in finding for each neighbor node the number of 2-hop neighbors for this node. This does not depend only on the number of neighbor nodes n but also on the two hop neighborhood. To get an evaluation only depending on n we will assume that the graph has a degree bounded by n. In such a case it is easy to show that this task can be done in less than n^2 steps. This evaluation corresponds to a worst case. Thus MPR selection is actually upper-bounded by $0(n^2)$. The complexity of the convex hull algorithm is in mean $O(nln(n))$. Thus from the complexity point of view, the convex hull approach seems better than the MPR method. However the given complexity of the convex hull is given for the

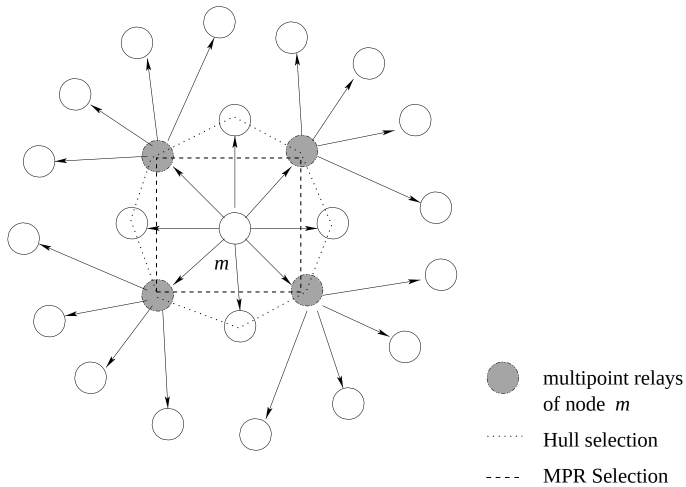

Figure 4: Different selections.

mean case while the MPR selection complexity is for the worst case. It can be thought that that the complexity of the two algorithms is actually similar. Moreover in a MANET (Mobile Ad Hoc NETwork) using OLSR as routing protocol, the MPR selection is already made by the routing protocol.

4 Simulation Results

The Wireless Mesh Network (WMN) comprises 100 nodes randomly located in a square of 1000 m $\times$ 1000 m. Self-locating nodes and simple nodes are also randomly elected. The maximum theoretical transmission range R_{max} was set to 170 m. The estimated position is obtained by a simple centroid formula. The simulations were performed under Java.

4.1 Position accuracy for a given number of anchors

We compare the accuracy of the three different described selection methods : MPR, convex hull and greedy. The simulation provides the average accuracy of the estimated position for each selection method : MPR, hull and greedy. During the simulation and for a given node we keep track of the number of neighbors with their type: #SLN denotes the number of its self-locating neighbor nodes and #SN denotes the number of its simple neighbor nodes [1].

[1] We assume that that these simple neighbor nodes have already an inferred position.

By running a large number of simulations, we obtain the average accuracy of the estimated position of this given node for each combination of number of self-locating neighbor nodes and of simple neighbor nodes (#SLN,#SN) for the three selection algorithms. The results of these extensive simulations are given in Figure 5. The relative confidence intervall of average position accuracy is 0.1% with a confidence of 95%. Thus the results shown in Figure 5 are meaningfull and error bars can not be seen.

In Figure 5 the graphs plot in (a),(c),(e) (respectively (b),(d),(f)) the average accuracy of the estimated position as a function of the number of simple nodes (resp. self-locating nodes) in the neighborhood. The number of self-locating nodes (resp. simple nodes) used by the position estimation process is set to 0 for figures (a) and (b). The number of self-locating nodes (resp. simple nodes) used by the position estimation process is set to 2 for figures (c) and (d) and the number of self-locating nodes (resp. simple nodes) used by the position estimation process is set to 4 for figures (e) and (d). As could be expected, we observe that the average position accuracy increases with the number of anchors. We also observe that the average position accuracy is better for a given number of self-locating nodes than for the same given number of simple nodes. The simulation confirms that the more precise the position information retrieved from the neighborhood is, the greater the accuracy of the estimated position.

Figure 5 shows that the selection algorithms significantly increases the position accuracy compared to the greedy approach. The convex hull selection increases the position accuracy by up to 20%, compared to the greedy approach. Figure 5 also shows that in every case, the MPR selection gives a better position accuracy than the convex hull approach, up to 15%. Thus the MPR selection can increase the position accuracy of a node by up to 45% compared to the greedy algorithm.

It can also be noticed in Figure 5 that MPR plots are only provided up to a given number of anchors. This is due to the fact that, in our simulations, a node has got a mean number of $\mathcal{N}_{th} = 10$ neighbors. With such a node density the MPR set generally contains seven or less nodes.

4.2 Position accuracy without considering the number of anchors

In Figure 6 we compare the greedy, the hull and the MPR algorithm for a given node in the network, we no longer consider the number of anchors as the parameter to compare the algorithms. We see that except for a very small density the three algorithms offer similar performances. We can even see that when the network nodes have more than 10 neighbors the MPR algorithm offers slightly better performances. This is interesting since this number of neighbors has been shown to be the lower limit above which a Wireless Mesh Network (WMN) is connected, see [27].

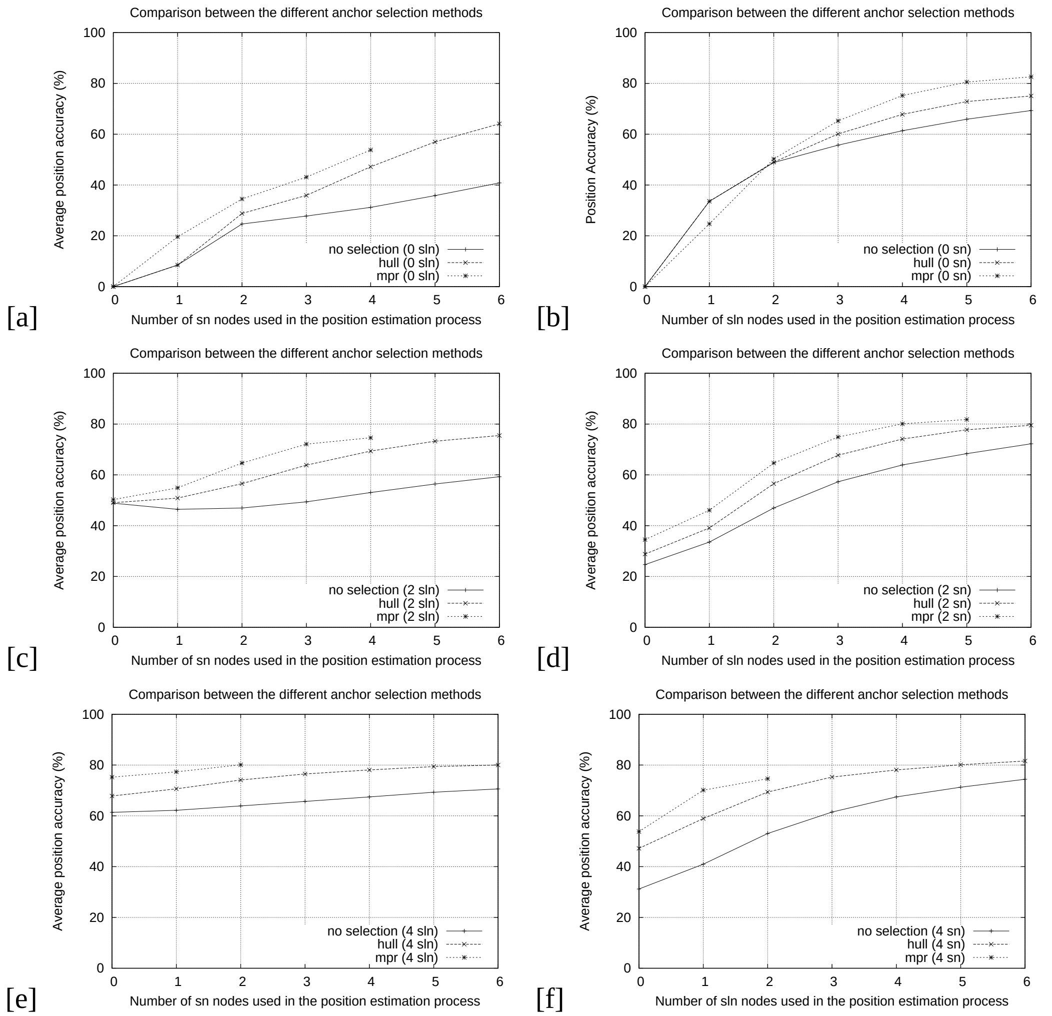

Figure 5: Impact of the anchor selection method on the average accuracy of the estimated position.

Of course the number of selected anchors in each algorithm is quite different. Figure 7 compares the number of selected nodes for the MPR selection, the convex hull and the greedy algorithm as a function of the mean number of neighbor node $\mathcal{N}_{th}$. For $\mathcal{N}_{th} = 10$ the greedy algorithm selects 1.5 more anchors than the hull algorithm and 4 more anchors than the MPR algorithm.

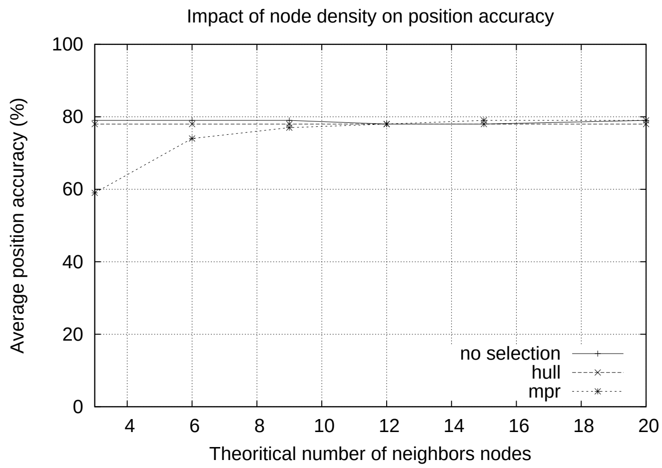

Figure 6: Comparison of the algotithm without constraint on the number of anchors.

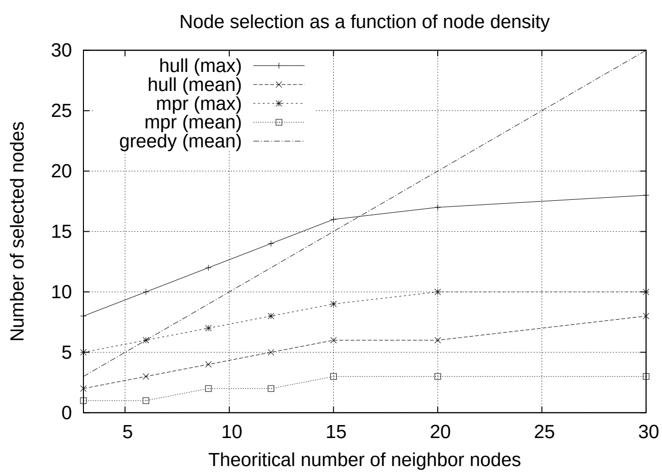

Figure 7: Number of selected nodes versus node density.

It can be seen from the simulation that most of the MPR set form a subset of the Hull nodes, see Figure 8 for an example given by the simulation. In Figure 9 we show the percentage of inclusion of MPR nodes in the Hull nodes as a function of the node density. For a node density leading to a maximum of 30 neighbors, the percentage of inclusion is above 90%. This result is somewhat surprising since MPR selection considers the connectivity metric to elect nodes whereas the convex hull selection only considers the distance. Note however that it is easy to construct examples for which $\{MPR\} \not\subset \{Hull\}$ but the probability of such a node configuration is small in a random network.

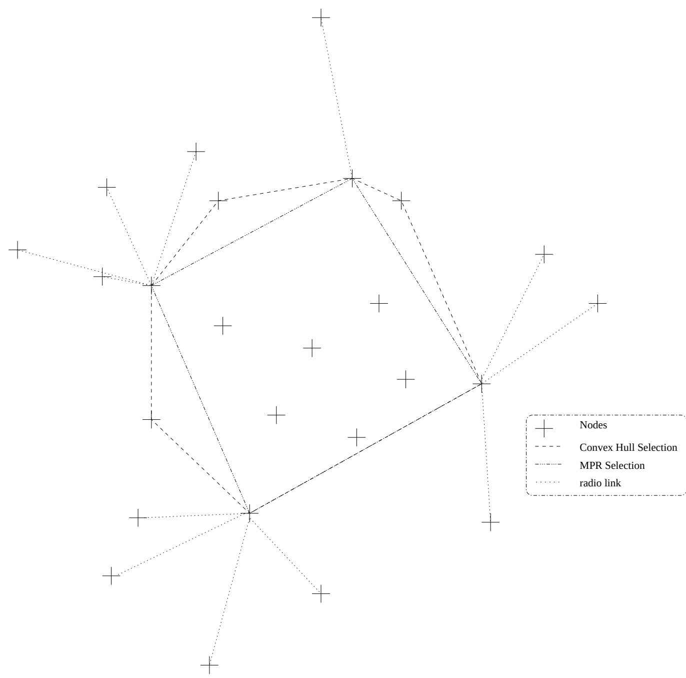

Figure 8: Random simulation topology case example.

5 Conclusion

In this paper, we have presented the use of MPR nodes to estimate the position of nodes in a wireless (mesh) network. Using such a technique the position accuracy, for a given number of anchors, is increased by up to 45% compared to the case where no selection is made (greedy algorithm).

When we compare the greedy, the hull and the MPR algorithms without constraining the number of anchors, we find similar position accuracy while the number of selected anchors is 4 times greater for the greedy algorithm than for the the MPR algorithm and 1.5 times greater than for the hull algorithm. We also show that the MPR nodes set is very often a subset of the convex hull nodes set. If geographical information is needed in an

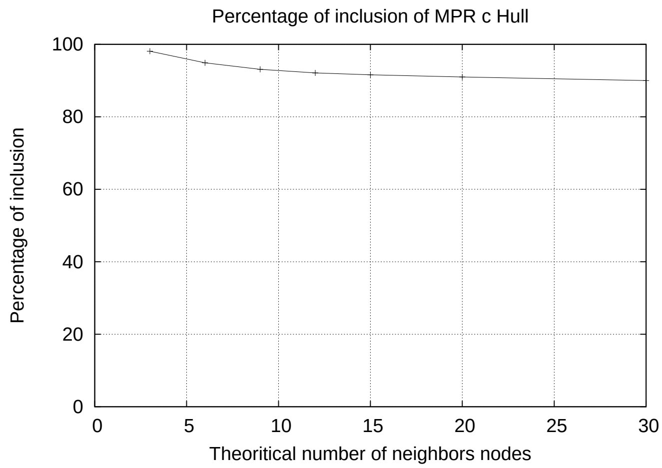

Figure 9: Percentage of inclusion MPR nodes into Hull nodes.

OLSR wireless mesh networks, the genuine OLSR MPR optimization is particularly useful to select anchors to estimate node position while the control OLSR message can be easily used to exchange information on the nodes' positions in the network.

We will focus our future work on determining the interesting uses of geographical information for OLSR networks.

References

[1] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, "Optimized link-state routing protocol," RFC 3626, October 2003.

[2] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System : Theory and Practice*. Springer-Verlag, 1997.

[3] C. Savarese, J. M. Rabaey, and J. Beutel, "Localization in distributed ad-hoc wireless sensor networks," *Proceedings of the ICASSP*, May 2001.

[4] J. Beutel, "Geolocalisation in a picoradio environment," Master's thesis, ETH Zurich, Electronics Lab, 1999.

[5] S. Capkun, M. Hamdi, and J. P. Hubaux, "Gps-free positioning in mobile ad-hoc networks," *Cluster Computing*, vol. 5, no. 2, April 2002.

[6] L. Doherty, K. S. J. Pister, and L. E. Ghaoui, “Convex optimization methods for sensor node position estimation,” in *Proceedings of IEEE INFOCOM’2001*, Anchorage, April 2001.

[7] S. Guha, R. Murty, and E. Sirer, “Sextant: A unified node and event localization framework using non-convex constraints,” Urbana-Champaign, IL, USA, May 2005, pp. 205–216.

[8] E. Ermel, A. Fladenmuller, G. Pujolle, and A. Cotton, “Improved position estimation in wireless heterogeneous networks,” in *Networking 2004*, May 2004.

[9] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-demand Distance Vector (AODV) routing,” July 2003, rFC 3561.

[10] D. B. Johnson, D. A. Maltz, and Y.-C. Hu, “The Dynamic Source Routing protocol for mobile ad hoc networks (DSR),” July 19 2004, internet-Draft, `draft-ietf-manet-dsr-10.txt`, work in progress.

[11] R. Ogier, F. Templin, and M. Lewis, “Topology dissemination Based on Reverse-Path Forwarding (TBRPF),” February 2004, rFC 3684.

[12] “Broadband Radio Access Networks (BRAN); HIgh PErformance Radio Local Area Network (HIPERLAN) type 1; functional specification,” ETSI, Tech. Rep. EN 300 652 ref. REN/BRAN-10-01, 1998.

[13] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, “Optimized link-state routing protocol,” RFC 3626, October 2003.

[14] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, 1988.

[15] L. Viennot, “Complexity results on election of multipoint relays in wireless networks,” INRIA, Tech. Rep. RR-3584, 1998.

[16] A. Qayyum, L. Viennot, and A. Laouiti, “Multipoint relaying technique for flooding broadcast messages in mobile wireless networks,” January 2002.

[17] D. R. Chand and S. S. Kapur, “An algorithm for convex polytopes,” *J. ACM*, vol. 17, no. 1, pp. 78–86, 1970.

[18] F. P. Preparata and S. J. Hong, “Convex hulls of finite sets of points in two and three dimensions,” *Commun. ACM*, vol. 20, no. 2, pp. 87–93, 1977.

[19] M. I. Shamos, "Computational geometry," Ph.D. dissertation, Department of Computer Science, Yale University, (USA), 1978.

[20] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*. Springer Verlag, 1991.

[21] R. Graham, "An efficient algorithm for determining the convex hull of a planar set," *Information Processing Letters*, vol. 1, pp. 132–133, 1972.

[22] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.

[23] F. Aurenhammer, "Voronoi diagrams – a survey of a fundamental geometric data structure," *ACM Computing Surveys*, vol. 23, no. 3, pp. 345–405, 1991.

[24] M. de Berg, M. Kreveld, M. Overmars, and O. Scharzkopf, *Computational Geometry, Algoritms and Application*. Springer, 1997.

[25] J. O'Rourke, *Computational Geometry in C*. Cambridge University Press, 1998.

[26] C. Lemaire, "Triangulatation de delaunay et arbres multidimensionnels," Ph.D. dissertation, Ecole des mines de Saint-Etienne, 1997.

[27] D. M. Blough, M. Leoncini, G. Resta, and P. Santi, "The lit k-neigh protocol for symmetric topology control in ad hoc networks," in *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. New York, NY, USA: ACM Press, 2003, pp. 141–152.

Unit´e de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifi que,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399